Rebuilding for Array Codes in Distributed Storage Systems

Zhiying Wang
Electrical Engineering Department
California Institute of Technology
Pasadena, CA 91125
Email: zhiying@caltech.edu

Alexandros G. Dimakis
Electrical Engineering Department
University of Southern California
Los Angeles, CA 90089-2560
Email: dimakis@usc.edu

Jehoshua Bruck
Electrical Engineering Department
California Institute of Technology
Pasadena, CA 91125
Email: bruck@caltech.edu

***Abstract*—In distributed storage systems that use coding, the issue of minimizing the communication required to rebuild a storage node after a failure arises. We consider the problem of repairing an erased node in a distributed storage system that uses an EVENODD code. EVENODD codes are maximum distance separable (MDS) array codes that are used to protect against erasures, and only require XOR operations for encoding and decoding. We show that when there are two redundancy nodes, to rebuild one erased systematic node, only $3/4$ of the information needs to be transmitted. Interestingly, in many cases, the required disk I/O is also minimized.**

I. Introduction

Coding techniques for storage systems have been used widely to protect data against errors or erasure for CDs, DVDs, Blu-ray Discs, and SSDs. Assume the data in a storage system is divided into packets of equal sizes. An (n, k) block code takes k information packets and encodes them into a total of n packets of the same size. Among coding schemes, maximum distance separable (MDS) codes offer maximal reliability for a given redundancy: any k packets are sufficient to retrieve all the information. Reed-Solomon codes [1] are the most well known MDS codes that are used widely in storage and communication applications. Another class of MDS codes are MDS array codes, for example EVENODD [2] and its extension [3], B-code [4], X-code [5], RDP [6], and STAR code [7]. In an array code, each of the packets consists of a column of elements (one or more binary bits), and the parities are computed by XORing some information bits. These codes have the advantage of low computational complexity over RS codes because the encoding and decoding only involve XOR operations.

Distributed storage systems involving storage nodes connected over networks have recently attracted a lot of attention. MDS codes can be used for erasure protection in distributed storage systems where encoded information is stored in a distributed manner. If no more than $n-k$ storage nodes are lost, then all the information can still be recovered from the surviving packets. Suppose one packet is erased, and instead of retrieving the entire k packets of information, if we are only interested in repairing the lost packet, then what is smallest amount of transmission needed (called the *repair bandwidth*)? If we transmit k packets from the other nodes to the erased one, then by the MDS property, we can certainly repair this node. But can we transmit less than k packets? More generally, if no more than $n-k$ nodes are erased, what is the repair bandwidth? This *repair problem* was first raised in [8], and was further studied in several works (e.g. [9]-[14]). A recent survey of this problem can be found in [15]. In [8], a cut-set lower bound for repair bandwidth is derived and in [11][12][13], this lower bound is matched for exact repair by code constructions for $k = 2, 3$, $n-1$ and $2k \leq n$. All of these constructions however require large finite fields. Very recently it was established that the cut-set bound of [8] is achievable for all values of k and n, [13][14]. However, the proof is theoretical and is based on very large finite fields. Hence, it does not provide the basis for constructing practical codes with small finite fields and high rate.

In this paper we take a different route: rather than trying to construct MDS codes that are easily repairable, we try to find ways to repair existing codes and specifically focus on the families of MDS array codes. A related and independent work can be found in [16], where single-disk recovery for RDP code was studied, and the recovery method and repair bandwidth is indeed similar to our result. Besides, [16] discussed balancing disk I/O reads in the recovery. Our work discusses the recovery of single or double disk recovery for EVENODD, X-code, STAR, and RDP code.

If the whole data object stored has size M bits, repairing a single erasure naively would require communicating (and reading) M bits from surviving storage nodes. Here we show that a single failed systematic node can be rebuilt after communicating only $\frac{3}{4}M+O(M^{1/2})$ bits. Note that the cut-set lower bound [8] scales like $\frac{1}{2}M + O(M^{1/2})$, so it remains open if the repair communication for EVENODD codes can be further reduced. Interestingly our repair scheme also requires significantly less disk I/O reads compared to naively reading the whole data object.

The rest of this paper is organized as follows. In Section II, we are going to define EVENODD code and the repair problem. Then the repair of one lost node is presented in Section III for EVENODD ($k = n-2$) and

978-1-4244-8865-0/10/$26.00 ©2010 IEEE

in Section IV for the extended EVENODD ($k < n-2$). In Section V, we consider the case with two erased nodes and $k = n-3$. At last, conclusion is made in Section VI.

II. Definitions

An $R \times n$ array code contains R rows and n columns (or packets). Each element in the array can be a single bit or a block of bits. We are going to call an element a *block*. In an (n, k) array code, k information columns, or systematic columns, are encoded into n columns. The *total amount of information* is $M = Rk$ blocks.

An EVENODD code [2] is a binary MDS array code that can correct up to 2 column erasures. For a prime number $p \geq 3$, the code contains $R = p-1$ rows and $n = p+2$ columns, where the first $k = p$ columns are information and the last two are parity. And the information is $M = (p-1)p$ blocks.

We will write an EVENODD code as:

$$\begin{array}{cccccc} a_{1,1} & a_{1,2} & \dots & a_{1,p} & b_{1,0} & b_{1,1} \\ a_{2,1} & a_{2,2} & \dots & a_{2,p} & b_{2,0} & b_{2,1} \\ \vdots & \vdots & & \vdots & \vdots & \vdots \\ a_{p-1,1} & a_{p-1,2} & \dots & a_{p-1,p} & b_{p-1,0} & b_{p-1,1} \end{array}$$

And we define an imaginary row $a_{p,j} = 0$, for all $j = 1, 2, \dots, p$, where 0 is a block of zeros. The *slope 0* or *horizontal parity* is defined as

$$b_{i,0} = \sum_{j=1}^{p} a_{i,j} \tag{1}$$

for $i = 1, \dots, p-1$. The addition here is bit-by-bit XOR for two blocks. A parity block of *slope* v, $-p < v < p$ and $v \neq 0$ is defined as

$$b_{i,v} = \sum_{j=1}^{p} a_{j,<i+v(1-j)>} + S_v = \sum_{j=1}^{p} a_{<i+v(1-j)>,j} + S_v \tag{2}$$

where $S_v = a_{p,1} + a_{p-v,2} + \cdots + a_{<p+v>,p} = \sum_{j=1}^{p} a_{<v(1-j)>,j}$ and $< x >= (x-1) \bmod p + 1$. Sometimes we omit the "<>" notation. When $v = 1$, we call it the *slope 1*, or *diagonal parity*. In EVENODD, parity columns are of slopes 0 and 1.

A similar code is RDP [6], where $R = p-1$, $n = p+1$, and $k = p-1$, for a prime number p. The diagonal parity sums up both the corresponding information blocks and one horizontal parity block. Another related code is X-code [5], where the parity blocks are of slope -1 and 1, and are placed as two additional rows, instead of two parity columns.

The code in [3] extended EVENODD to more than 2 columns of parity. This code has $n = p + r$, $k = p$, and $R = p-1$. The information columns are the same as EVENODD, but r parity columns of slopes $0, 1, \dots, r-1$ are used. It is shown in [3] that such a code is MDS when $r \leq 3$ and conditions for a code to be MDS are derived for $r \leq 8$.

STAR code [7] is an MDS array code with $k = p$, $R = p-1$, $n = p+3$, and the parity columns are of slope 0, 1, and -1.

A *parity group* $B_{i,v}$ of slope v contains a parity block $b_{i,v}$ and the information blocks in the sum in equations (1) (2), $i = 1, 2, \dots, p-1$. S_v is considered as a single information block. If $v = 0$, it is a *horizontal parity group*, and if $v = 1$, we call it a *diagonal parity group*.

By (1), each horizontal parity group $B_{i,0}$ contains $a_{i,<k+1-i>} \in B_{k,1}$, for all $k = 1, 2, \dots, p-1$. So we say $B_{i,0}$ *crosses with* $B_{k,1}$, for all $k = 1, 2, \dots, p-1$. Conversely, each diagonal parity group $B_{i,1}$ contains $a_{k,<i+1-k>} \in B_{k,0}$, for all $k = 1, 2, \dots, p-1$. Therefore, $B_{i,1}$ crosses with $B_{k,0}$ for all $k = 1, 2, \dots, p-1$. The shared block of two parity groups is called the *crossing*. Generally, two parity groups $B_{i,v}$ and $B_{k,u}$ cross, for $v \neq u$, $1 \leq i, k \leq p-1$. If they cross at $a_{p,<i+v>} = 0$, we call it a *zero crossing*. A zero crossing does not really exist since the p-th row is imaginary. A zero crossing occurs if and only if

$$u, v \neq 0 \text{ and } < i+v >=< k+u > \tag{3}$$

Moreover, each information block belongs to only one parity group of slope v.

Suppose the n packets are stored in n different nodes in a connected network. Each storage node contains exactly one packet (or one column). Assume $n-d$ nodes are erased, $d \geq k$. Suppose we recover the nodes successively. For any specified erased node, how many blocks from the other storage nodes are needed to recover it? We can either send data in a single block, or a linear combination of several blocks in one node, both of which are counted as one block of transmission. The total number of blocks transmitted to recover the specified node is called the *repair bandwidth* γ. The *repair problem* for distributed storage system asks what the smallest γ is, for fixed M, d, k. In [8], a cut-set lower bound is derived (and is achieved only when each node transmits the same number of blocks):

$$\gamma^* = \frac{Md}{k(d-k+1)} \tag{4}$$

In this paper, we use MDS array codes as distributed storage codes. We will give repair methods and compute the corresponding bandwidth γ.

Example 1. *Consider the EVENODD code with $p = 3$. Set $a_{1,3} = a_{2,3} = 0$ for all codewords, then the code will contain only 2 columns of information. The resulting code is a $(4, 2)$ MDS code and this is called* shortened *EVENODD (see Figure 1). It can be verified that if any node is erased, then sending 1 block from each of the other nodes is sufficient to recover it. And this actually matches the bound (4). Figure 1 shows how to recover the first or the fourth column. Notice that a sum block is sent in some cases. For instance, to recover the first column, the sum $b_{1,1} + b_{2,1}$ is sent from the fourth column.*

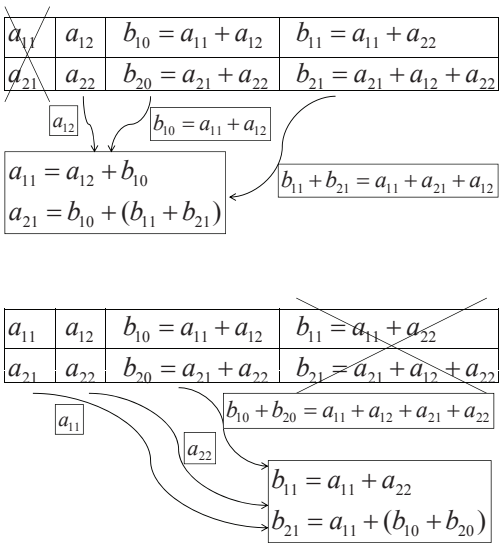

Fig. 1. Repair of a $(4, 2)$ EVENODD code if the first column (top graph) or the fourth column (bottom graph) is erased. In both cases, three blocks are transmitted.

In this paper, shortening of a code is not considered and we will focus on the recovery of systematic nodes, given that 1 or 2 systematic nodes are erased. And we send no linear combinations of data except the sum $\sum_{i=1}^{p-1} b_{i,v}$ from the parity node of slope v, for all v defined in an array code. In addition, we assume that each node can transmit a different number of blocks.

III. Repair for Codes with 2 Parity Nodes

First, let us consider the repair problem of losing one systematic node, $n - d = 1$, and $n - k = 2$. We will use EVENODD to explain the repair method, and the recovery will be very similar if RDP or X-code is considered.

By the symmetry of the code, we assume that the first column is missing. Each block in the first column must be recovered through either the horizontal or the diagonal parity group including this block. Suppose we use x horizontal parity groups and $p - 1 - x$ diagonal parity groups to recover the column, $0 \le x \le p - 1$. These parity groups include all blocks of the first column exactly once.

Notice that $S_1 = \sum_{i=1}^{p-1} b_{i,0} + \sum_{i=1}^{p-1} b_{i,1}$, so we can send $\sum_{i=1}^{p-1} b_{i,0}$ from the $(p+1)$-th node, and $\sum_{i=1}^{p-1} b_{i,1}$ from the $(p+2)$-th node, and recover S_1 with 2 blocks of transmission. For the discussion below, assume S_1 is known.

For each horizontal parity group $B_{i,0}$, we send $b_{i,0}$ and $a_{i,j}$, $j = 2, 3, \ldots, p$. So we need p blocks. For each diagonal parity group $B_{i,1}$, as S_1 is known, we send $b_{i,1}$ and $a_{j,<i+1-j>}$, $j = 1, 2, \ldots, i-1, i+1, \ldots, p-1$, which is $p - 1$ blocks in total.

If two parity groups cross at one block, there is no need to send this block twice. As shown in Section II, any horizontal and any diagonal parity group cross at a block, and each block can be the crossing of two groups at most once. There are $x(p-1-x)$ crossings. The total number of blocks sent is

$$\begin{aligned}\gamma &= \underbrace{xp}_{\text{horizontal}} + \underbrace{(p-1-x)(p-1)}_{\text{diagonal}} + \underbrace{2}_{S_1} - \underbrace{x(p-1-x)}_{\text{crossings}} \\ &= (p-1)p + 2 - (x+1)(p-1-x) \qquad (5) \\ &\ge (p-1)p + 2 - (p^2-1)/4 = (3p^2 - 4p + 9)/4\end{aligned}$$

The equality holds when $x = (p-1)/2$ or $x = (p-3)/2$, where x is an integer.

This result states that we only need to send about $3/4$ of the total amount of information. And the slopes of the n chosen parity groups do not matter as long as half are horizontal and half are diagonal. Moreover, similar repair bandwidth can be achieved using RDP or X-code. For RDP code, the repair bandwidth is $\frac{3(p-1)^2}{4}$ [17], which was also derived independently in [16]. For X-code, the repair bandwidth is at most $\frac{3p^2-2p+5}{4}$ [17].

Also, equation (5) is optimal in some conditions:

Theorem 2. *The transmission bandwidth in (5) is optimal to recover a systematic node for EVENODD if no linear combinations are sent except $\sum_{i=1}^{p-1} b_{i,v}$, for $v = 0, 1$.*

Proof: To recover a systematic node, say, the first node, parity blocks $b_{i,v}$, $i = 1, 2, \ldots, p-1$ must be sent, where v can be 0 or 1 for each i. This is because $a_{i,1}$ is only included in $b_{i,0}$ or $b_{i,1}$. Besides, given $b_{i,v}$, the whole parity group $B_{i,v}$ must be sent to recover the lost block. Therefore, our strategy of choosing x horizontal parity groups and $p - 1 - x$ diagonal parity groups has the most efficient transmission. Finally, since (5) is minimized over all possible x, it is optimal. ■

The lower bound by (4) is

$$\frac{Md}{(d-k+1)k} = \frac{M(n-1)}{(n-k)k} = \frac{p(p-1)(p+1)}{2p} = \frac{p^2-1}{2}$$

where $d = n - 1$, $n = p + 2$, $k = p$, and $M = p(p-1)$. It should be noted that (4) assumes that each node sends the same number of blocks, but our method does not.

Example 3. *Consider the EVENODD code with $p = 5$ in Figure 2. For $1 \le i \le 4$, the code has information blocks $a_{i,j}$, $1 \le j \le 5$, and parity blocks $b_{i,v}$, $v = 0, 1$. Suppose the first column is lost. Then by (5), we can choose parity groups $B_{1,0}, B_{2,0}, B_{3,1}, B_{4,1}$. The blocks sent are: $\sum_{i=1}^{p-1} b_{i,0}, \sum_{i=1}^{p-1} b_{i,1}, b_{1,0}, b_{2,0}, b_{3,1}, b_{4,1}$ from the parity nodes and $a_{1,2}, a_{1,3}, a_{1,4}, a_{1,5}, a_{2,2}, a_{2,3}, a_{2,4}, a_{2,5}, a_{4,5}, a_{3,2}$ from the systematic nodes. Altogether, we send 16 blocks, the number specified by (5). We can see that $a_{1,3}$ is the crossing of $B_{1,0}$ and $B_{3,1}$. Similarly, $a_{1,4}, a_{2,2}, a_{2,3}$ are crossings and are only sent once for two parity groups.*

IV. r Parity Nodes and One Erased Node

Next we discuss the repair of array codes with r columns of parity, $r \ge 3$. And we consider the recovery in the case of one missing systematic column. In this section, we are going to use the extended EVENODD

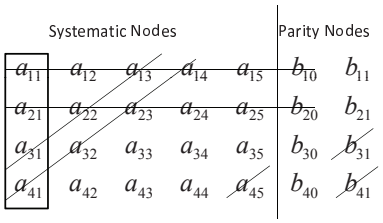

Fig. 2. Repair of an EVENODD code with $p = 5$. The first column is erased, shown in the box. 14 blocks are transmitted, shown by the blocks on the horizontal or diagonal lines. Each line (with wrap around) is a parity group. 2 blocks in summation form, $\sum_{i=1}^{p-1} b_{i,0}, \sum_{i=1}^{p-1} b_{i,1}$ are also needed but are not shown in the graph.

code [3] with parity columns of slopes $0, 1, \ldots, r-1$. Similar results can be derived for STAR code. Suppose the first column is erased without loss of generality.

Let us first assume $r = 3$, so the parity columns have slopes $0, 1, 2$. The repair strategy is: sending parity groups $B_{3n+v,v}$ for $v = 0, 1, 2$ and $1 \le 3n + v \le p - 1$.

Let $A = \lfloor (p-1)/3 \rfloor$, $(p-4)/3 < A \le (p-1)/3$. Then $0 \le n \le A$ and each slope has no more than $\lceil (p-1)/3 \rceil$ but no less than $\lfloor (p-1)/3 \rfloor = A$ parity groups.

Since there are three different slopes, there are crossings between slope 0 and 1, slope 1 and 2, and slope 2 and 0. For any two parity groups $B_{i,1}$ and $B_{k,2}$, $< k - i > \neq 1$, so (3) does not hold. Hence no zero crossing exists for the chosen parity groups. Hence, every crossing corresponds to one block of saving in transmission. However, the total number of crossings is not equal to the sum of crossings between every two parity groups with different slopes. Three parity groups with slopes 0, 1, and 2 may share a common block, which should be subtracted from the sum.

Notice that the parity group $B_{i,v}$ contains the block $a_{i-vy,y+1}$. The modulo function "<>" is omitted in the subscripts. For three transmitted parity groups $B_{3n,0}, B_{3m+1,1}, B_{3l+2,2}$, if there is a common block in column $y + 1$, then it is in row $3n \equiv 3m + 1 - y \equiv 3l + 2 - 2y \pmod p$. To solve this, we get $y \equiv 3(m - n) + 1 \equiv 3(l - m) + 1 \pmod p$, or $m - n \equiv l - m \pmod p$. Notice $0 \le n, m, l < p/3$, so $-p/3 < m - n, l - m < p/3$. Therefore, $m - n = l - m$ without modulo p. Thus $l - n$ must be an even number. For fixed n, either $n \le m \le l \le A$, and there are no more than $(A - n)/2 + 1$ solutions for (m, l); or $0 \le l < m < n$, and the number of (m, l) is no more than $n/2$. Hence, the number of (n, m, l) is no more than $\sum_{n=1}^{A}((A - n)/2 + 1 + n/2) = A^2/2 + A$.

The total number of blocks in the $p - 1$ chosen parity groups is less than $p(p - 1)$. There are no less than A parity groups of slope v, for all $0 \le v \le 2$, therefore for $0 \le u < v \le 2$, parity groups with slopes u and v have no less than A^2 crossings. Hence the total number of blocks sent in order to recover one column is:

$$\begin{aligned} \gamma &< p(p-1) - 3A^2 + (A^2 + 2A)/2 + 3 \\ &< 13p^2/18 + 17p/9p - 47/18 \end{aligned} \quad (6)$$

By abuse of notation, we write $B_{m,v} = \{a_{<m+v(1-j)>,j} : j = 2, \ldots, p\}$, $1 \le m \le p - 1$, $0 \le v \le r - 1$. Let $M_v \subseteq \{1, 2, \ldots, q - 1\}$ be disjoint sets such that $\cup_{v=0}^{r-1} M_v = \{1, 2, \ldots, q - 1\}$. Let $B_{M_v,v} = \cup_{m \in M_v} B_{m,v}$. For given M_v, define a function f as $f(v_1, v_2, \ldots, v_k) = |\{m_1 \in M_{v_1}, \ldots, m_k \in M_{v_k} : (m_2 - m_1)/(v_2 - v_1) \equiv \ldots (m_k - m_{k-1})/(v_k - v_{k-1}) \bmod p\}|$, for $k \ge 3$, and $0 \le v_1 < \cdots < v_k \le r - 1$. Then we have

Theorem 4. *For the extended EVENODD with $r \ge 3$, the repair bandwidth for one erased systematic node is* $\gamma < p(p-1) + p + r - \sum_{0 \le v_1 < v_2 < r-1} |M_{v_1}||M_{v_2}| + \sum_{0 \le v_1 < v_2 < v_3 \le r-1} f(v_1, v_2, v_3) - \cdots + (-1)^{r-1} f(0, 1, \ldots, r-1)$

The proof can be found in [17].

Define $M_v = \{rn + v : 1 \le rn + v \le p - 1\}$, for $v = 0, 1, \ldots, r - 1$. If $r = 3$, (6) is a special case of the theorem and $\gamma \approx 13M/18$. If $r = 4, 5$, $\gamma \approx 17M/24, 53M/75$, respectively [17].

V. 3 Parity Nodes and 2 Erased Nodes

Up to now, we have considered the recovery problem given that one column is erased. Next, let us assume that two information columns are erased and we need to recover them successively. So we first recover one of the erased nodes, and then the other one. The first recovery is discussed in this section, and the second recovery was already discussed in the previous sections. Suppose we have 3 columns of parity with slopes -1, 0, and 1, which is in fact the STAR code in [7]. Again, the arguments can be applied to extended EVENODD in a similar way. Without loss of generality, assume the first and $(x+1)$-th columns are missing, $1 \le x \le p - 1$.

Let $B_{i,0}$,$B_{i,1}$, and $B_{i,-1}$ be i-th parity group of slopes 0, 1, and -1, respectively, $i = 1, 2, \ldots, p - 1$. The following are $3(p-1)/2$ parity groups that repair the first column: $B_{0,-1}, B_{x,0}, B_{2x,1}, B_{2x,-1}, B_{3x,0}, B_{4x,1}, \ldots, B_{(p-3)x,-1}, B_{(p-2)x,0}, B_{(p-1)x,1}$. For each parity block above, the corresponding recovered blocks are: $a_{x,1+x}$, $a_{x,1}, a_{2x,1}, a_{3x,1+x}, a_{3x,1}, a_{4x,1}, \ldots, a_{(p-2)x,1+x}$, $a_{(p-2)x,1}, a_{(p-1)x,1}$. An example of $p = 5, x = 1$ is shown in Figure 3.

Rearrange the columns in the following order: Columns $1, 1 + x, 1 + 2x, \ldots, 1 + (p - 1)x$ (every index is computed modulo p). We can see that the chosen parity groups $B_{jx,0}$, $j = x, 3x, \ldots, (p - 2)x$ contain the blocks in Rows $Z = \{x, 3x, \ldots, (p - 2)x\}$. $B_{jx,1}$ contains blocks $a_{jx,1}, a_{(j-1)x,1+x}, \ldots, a_{(j-p+1)x,1+(p-1)x}$, for $j = 2, 4, \ldots, p - 1$. And similarly $B_{jx,-1}$ contains blocks $a_{jx,1}, a_{(j+1)x,1+x}, \ldots, a_{(j+p-1)x,1+(p-1)x}$, for $j = 0, 2, \ldots, p - 3$.

Now notice that the blocks included in the above parity groups have the $(1 + x)$-th column as the vertical symmetry axis. That is, the row indices of the blocks needed in Columns 1 and $1 + 2x$ are the same; those of Columns $1 + (p - 1)x$ and $1 + 3x$ are the same; ...; those

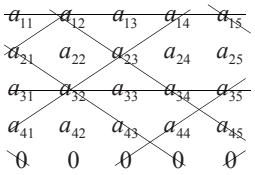

Fig. 3. The recovery strategy for the first column in STAR code when the first and second columns are missing. $p = 5$, $x = 1$.

of Columns $1 + (p + 3)x/2$ and $1 + (p + 1)x/2$ are the same. For example, the second column in Figure 3 is the symmetry axis. Thus, we only need to consider Columns $1 + 2x, 1 + 3x, \ldots, 1 + (p + 1)x/2$.

For columns $1+ix$, where i is even and $2 \leq i \leq (p+1)/2$, parity groups $\{B_{2x,1}, B_{4x,1}, \ldots, B_{(p-1)x,1}\}$ include the blocks in Rows $X = \{2x, 4x, \ldots, (p-1-i)x\}$. And parity groups $\{B_{0,-1}, B_{2x,-1}, \ldots, B_{(p-3)x,-1}\}$ include the blocks in Rows $Y = \{ix, (i + 2)x, \ldots, (p - 1)x\}$. Since $2 \leq i \leq (p+1)/2$, we have $i \leq (p-1-i)+2$, and $X \cup Y = \{2x, 4x, \ldots, (p-1)x\}$. Hence $X \cup Y \cup Z = \{1, 2, \ldots, p - 1\}$. Thus every block in Column $1 + ix$ needs to be sent, for even i.

Similarly, for Columns $1 + ix$, where i is odd and $3 \leq i \leq (p + 1)/2$, parity groups $\{B_{2x,1}, B_{4x,1}, \ldots, B_{(p-1)x,1}\}$ include the blocks in Rows $X = \{(p - i + 2)x, (p - i + 4)x, \ldots, (p - 1)x\}$. Parity groups $\{B_{0,-1}, B_{2x,-1}, \ldots, B_{(p-3)x,-1}\}$ include the blocks in Rows $Y = \{2x, 4x, \ldots, (i - 3)x\}$. Since $2 \leq i \leq (p + 1)/2$, we have $i - 3 < p - i + 2$, and $X \cup Y = \{2x, 4x, \ldots, (i - 3)x, (p - i + 2)x, (p - i + 4)x, \ldots, (p - 1)x\}$. Therefore, the rows not included in X or Y or Z are $W = \{(i-1)x, (i+1)x, \ldots, (p-i)x\}$ and $|W| = (p + 3)/2 - i$. The total saving in block transmissions for all the columns is:

$$2 \sum_{i \text{ odd},\ 3 \leq i \leq (p+1)/2} (\frac{p+3}{2} - i) = \begin{cases} \frac{(p-1)^2}{8}, & \frac{p+1}{2} \text{ odd} \\ \frac{(p+1)(p-3)}{8}, & \frac{p+1}{2} \text{ even} \end{cases}$$

The above argument can be summarized as:

Theorem 5. *When two systematic nodes are erased in a STAR code, there exist a strategy that transmit about $7/8$ of all the information blocks, and about $1/2$ of all the parity blocks so as to recover one node.*

The repair bandwidth γ in the above theorem is about $7p^2/8$. Comparing it to the lower bound (4), $\frac{Md}{k(d-k+1)} = \frac{p(p-1)(p+1)}{2p} \approx \frac{p^2}{2}$, we see a gap of $\frac{3p^2}{8}$ in total transmission.

VI. Conclusions

We presented an efficient way to repair one lost node in EVENODD codes and two lost nodes in STAR codes. Our achievable schemes outperform the naive method of rebuilding by reconstructing all the data. For EVENODD codes, a bandwidth of roughly $3M/4$ is sufficient to repair an erased systematic node. Moreover, if no linear combinations of bits are transmitted, the proposed repair method has optimal repair bandwidth with the sole exception of the sum of the parity nodes. Since array codes only operate on binary symbols, and our repair method involves no linear combination of content within a node except in the parity nodes, the proposed construction is computationally simple and also requires smaller disk I/O to read data during repairs.

There are several open problems on using array codes for distributed storage. Although our scheme does not achieve the information theoretic cut-set bound, it is not clear if that bound is achievable for fixed code structures or limited field sizes. If we allow linear combinations of bits within each node, the optimal repair remains unknown. Our simulations indicate that shortening of EVENODD (using less than p columns of information) further reduces the repair bandwidth, but proper shortening rules and repair methods need to be developed. Repairing other families of array codes or Reed-Solomon codes would also be of substantial practical interest.

References

[1] I. Reed and G. Solomon. Polynomial codes over certain finite fields. *Journal of the SIAM*, 8(2):300–304, 1960.

[2] M. Blaum, J. Brady, J. Bruck, and J. Menon. EVENODD: an efficient scheme for tolerating double disk failures in raid architectures. *IEEE Trans. on Computers*, 44(2):192–202, 1995.

[3] M. Blaum, J. Bruck, and A. Vardy. MDS array codes with independent parity symbols. *IEEE Trans. on Information Theory*, 42(2):529–542, 1996.

[4] L. Xu, V. Bohossian, J. Bruck, and D. G. Wagner. Low-density MDS codes and factors of complete graphs. *IEEE Trans. on Information Theory*, 45(6):1817–1826, 1999.

[5] L. Xu and J. Bruck. X-Code: MDS array codes with optimal encoding. *IEEE Trans. on Information Theory*, 45(1):272–275, 1999.

[6] P. Corbett, B. English, A. Goel, T. Grcanac, S. Kleiman, J. Leong, and S. Sankar. Row-diagonal parity for double disk failure correction. In *Proc. of the 3rd USENIX Symposium on File and Storage Technologies (FAST '04)*, pages 1–14, 2004.

[7] C. Huang and L. Xu. STAR: an efficient coding scheme for correcting triple storage node failures. *IEEE Trans. on Computers*, 57(7):889–901, 2008.

[8] A. G. Dimakis, P. G. Godfrey, Y. Wu, M. J. Wainwright, and K. Ramchandran. Network coding for distributed storage systems. *IEEE Trans. on Information Theory*, to appear.

[9] Y. Wu, A. G. Dimakis, and K. Ramchandran. Deterministic regenerating codes for distributed storage. In *Allerton Conference on Control, Computing, and Communication*, 2007.

[10] Y. Wu. Existence and construction of capacity-achieving network codes for distributed storage. In *Proc. IEEE ISIT*, 2009.

[11] Y. Wu and A. G. Dimakis. Reducing repair traffic for erasure coding-based storage via interference alignment. In *Proc. IEEE ISIT*, 2009.

[12] D. Cullina, A. G. Dimakis, and T. Ho. Searching for minimum storage regenerating codes. In *Allerton Conference on Control, Computing, and Communication*, 2009.

[13] C. Suh and K. Ramchandran. Exact regeneration codes for distributed storage repair using interference alignment. In *Proc. IEEE ISIT*, 2010.

[14] V. R. Cadambe, S. A. Jafar, and H. Maleki. Distributed data storage with minimum storage regenerating codes - exact and functional repair are asymptotically equally efficient. http://arxiv.org/pdf/1004.4299.

[15] A. G. Dimakis, K. Ramchandran, Y. Wu, and C. Suh. A survey on network codes for distributed storage. http://arxiv.org/pdf/1004.4438.

[16] L. Xiang, Y. Xu, J. C.S. Lui, and Q. Chang. Optimal recovery of single disk failure in RDP code storage systems. *SIGMETRICS Performance Evaluation Review*, 38(1):119–130, 2010.

[17] Z. Wang, A. G. Dimakis, and J. Bruck. Rebuilding for array codes in distributed storage systems. http://www.paradise.caltech.edu/etr.html.